**Ejercicio XQUERY 1B**

Dado el siguiente código XML realiza las sentencias XQUERY necesarias para obtener los resultados que se indican.

Ejercicio4bookstore.xml

<?xml version="1.0" encoding="UTF-8"?>

<bookstore>

<book category="COOKING">

<title lang="en">Everyday Italian</title>

<author>Giada De Laurentiis</author>

<year>2005</year>

<price>30.00</price>

</book>

<book category="CHILDREN">

<title lang="en">Harry Potter</title>

<author>J K. Rowling</author>

<year>2005</year>

<price>29.99</price>

</book>

<book category="WEB">

<title lang="en">XQuery Kick Start</title>

<author>James McGovern</author>

<author>Per Bothner</author>

<author>Kurt Cagle</author>

<author>James Linn</author>

<author>Vaidyanathan Nagarajan</author>

<year>2003</year>

<price>49.99</price>

</book>

<book category="WEB">

<title lang="en">Learning XML</title>

<author>Erik T. Ray</author>

<year>2003</year>

<price>39.95</price>

</book>

</bookstore>

1. Mostrar cuántos libros hay, y etiquetarlo con "total".

let $libro\_count := count(/bookstore/book)

return <total>{$libro\_count }</total>

1. Mostrar los títulos de los libros y al final una etiqueta con el número total de libros.

let $libro\_count := count(/bookstore/book),

$libros := (

for $libro in /bookstore/book

return <libro>{$libro/title/text()}</libro>)

return <libros>

{$libros}

<total>{$libro\_count}</total>

</libros>

1. Mostrar el precio mínimo y máximo de los libros.

for $libro\_count in/bookstore

return <libros>

<max>{max($libro\_count/book/price)}</max>

<min>{min($libro\_count/book/price)}</min>

</libros>

1. Mostrar la suma total de los precios de los libros con la etiqueta "total".

for $libro\_count in/bookstore

return <libros>

<max>{max($libro\_count/book/price)}</max>

<min>{min($libro\_count/book/price)}</min>

</libros>

1. Mostrar cada uno de los precios de los libros, y al final una nueva etiqueta con la suma de los precios.

let $libros\_price := sum(/bookstore/book/price),

$libro\_price := (

for $libro in /bookstore

return <precio\_ind>{$libro/book/price}</precio\_ind>)

return <libros>

{$libro\_price}

<total>{$libros\_price}</total>

</libros>

1. Mostrar el título y el número de autores que tiene cada título en etiquetas diferentes.

for $libro in /bookstore/book,

$libro\_autores in (

let $autor := count($libro/author)

return <autores>{$autor}</autores>)

return <libro>

<title>{$libro/title/text()}</title>

{$libro\_autores}

</libro>

1. Mostrar en la misma etiqueta el título y entre paréntesis el número de autores que tiene ese título.

for $libro in /bookstore/book

return <libro>{$libro/title/text()} ({count($libro/author)})</libro>

1. Mostrar los libros escritos en años que terminen en "3".

for $libro in /bookstore/book

where ends-with($libro/year, "3")

return <libro>{$libro/title/text()}</libro>

1. Mostrar los libros cuya categoría empiece por "C".

for $libro in /bookstore/book

where starts-with($libro/@category, "C")

return <libro>{$libro/title/text()}</libro>

1. Mostrar los libros que tengan una "X" mayúscula o minúscula en el título.

for $libro in /bookstore/book

where contains($libro/title, "X")

return <libro>{$libro/title/text()}</libro>

1. Mostrar el título y el número de caracteres que tiene cada título, cada uno con su propia etiqueta.

for $libro in /bookstore/book

return <libro>

<titulo>{$libro/title/text()}</titulo>

<caracteres>{string-length($libro/title)}</caracteres>

</libro>

1. Mostrar todos los años en los que se ha publicado un libro eliminando los repetidos. Etiquétalos con "año".

for $libro in distinct-values(/bookstore/book/year)

return <año>{$libro}</año>

1. Mostrar todos los autores eliminando los que se repiten y ordenados por el número de caracteres que tiene cada autor.

for $autor in distinct-values(/bookstore/book/author)

order by string-length($autor)

return <año>{$autor}, ({string-length($autor)})</año>